Level 4
Diploma in Cyber Security

(DA701-02)

Qualification

Level 4 Diploma in Cyber Security **(DA701-01)**

Title and level	**GLH**	**TQT**	**Worl Global Skills Qualification number**
Level 4 Diploma in Cyber Security	600	1200	DA701-02

Introduction

This document tells you what you need to do to deliver the qualifications:

The GSQ Level 4 Diploma in Cyber Security is a qualification that is expected to be directed study, and each candidate is expected to achieve all learning outcomes to attain the full qualification.

The purpose of the qualification is to develop students' knowledge and understanding of aspects of management and other topics relevant to the tourism and hospitality industry.

Areas covered include:
- Information Technology
- Computer Networks
- Operating Systems
- Data Structures

Candidate Entry Requirements

Candidates should not be entered for a qualification of the same type, content and level as that of a qualification they already hold.

It is recommended that candidates should have a minimum of a level 3 qualification. However, experience of working in the industry at the level 3 standard will be adequate to ensure they have the potential to gain the qualification successfully.

Qualification Units

To achieve the **DA701-01 Level 4 Diploma in Cyber Security**

Mandatory Units

GSQ Unit code	Ofqual Unit Reference	Unit Title	Credit Value	GLH
818	**A/651/4218**	**Information Technology Security**	**20**	**100**
819	**D/651/4219**	**Principles of Computer Systems**	**20**	**100**
820	**J/651/4220**	**Algorithms and Data Structures**	**20**	**100**
821	**K/651/4221**	**Computer Networks**	**20**	**100**
822	**L/651/4222**	**Mathematics for Computer Science**	**20**	**100**
823	**M/651/4223**	**Operating Systems**	**20**	**100**

Total Qualification Time

Total Qualification Time (TQT) is the total amount of time, in hours, expected to be spent by a Learner to achieve a qualification. This includes both guided learning hours and hours spent in preparation, study and assessment.

Title and level	**GLH**	**TQT**
Level 4 Diploma in Cyber Security	600	1200

Portfolio of evidence

To achieve this portfolio, you will need to produce a portfolio of evidence to show that you have the knowledge and skills in your chosen qualification. You will need to demonstrate evidence that they have fulfilled all the learning outcomes and meet the standards specified by all assessment criteria. Judgement that the learners have successfully fulfilled the assessment criteria is made by the assessor.

It is the centres responsibility to produce assignment briefs to support the students with their assignments, these will include

- Projects
- Audio-visual media
- Evidence of prior learning or attainment
- Written questions
- Assignments
- Case studies

Centre Staffing

Staff delivering these qualifications must be able to demonstrate that they meet the following occupational expertise requirements. They should:

- be occupationally competent or technically knowledgeable in the area[s] for which they are delivering training and/or have experience of providing training. This knowledge must be to

the same level as the training being delivered
- have recent relevant experience in the specific area they will be assessing and continue to achieve and minimum 30 professional development hours per year
- have credible experience of providing training

Centre staff may undertake more than one role, e.g. tutor and assessor or internal quality assurer, but cannot internally quality assure their own assessments.

Tutors/Assessors and Internal Quality Assurer

Assessor/Internal Quality Assurer qualifications are valued as qualifications for Centre staff, but they are not currently a requirement for the qualifications.

Continual Professional Development (CPD)

Centres must support their staff to ensure that they have a current knowledge of the occupational area, that delivery, mentoring, training, assessment and verification is in line with best practice, and that it takes account of any national or legislative developments.

Qualification Structure

Unit 818 Information Technology Security
UAN: A/651/4218
Level: 4
Credit value: 20
GLH: 100

Aim

This unit will introduce the basic principles of security in digital systems - cyber security - explaining the role of cyber security in protecting digital systems, networks, software, and data from unauthorised access, theft, damage, or disruption. This unit will help learners to understand the nature of digital information and the importance of keeping information secure, as well as methods for doing so. As part of the unit, learners will develop an appreciation for the broad space of threats that cyber systems face, and will understand how these threats can be prevented, mitigated, and responded to with appropriate frameworks; practices and principles; and techniques such as encryption and hashing.

Learning outcome 1 1. Understand the nature of information and what is meant by information privacy.	1.1 1.2 1.3 1.4	Define information and give examples of information. Explain what is meant by information privacy. Analyse the value of information from contrasting perspectives. Analyse the importance of information privacy and the potential consequences when information is not kept private.
Learning outcome 2 2. Understand key concepts in information security, including threats and risks.	2.1 2.2 2.4 2.3	Describe the common types of information security threats for modern computing systems. Explain the Confidentiality, Integrity, and Availability (CIA Triad) framework and its meaning in the context of information security. Analyse the role of risk management in identifying security threats and vulnerabilities. Evaluate several key cyber security models and their focuses.
Learning outcome 3 3. Understand the importance of secure design in Cyber Security systems.	3.1 3.2 3.3	Describe the classes of common vulnerabilities in software and hardware systems. Explain the principles of secure design of cyber systems. Analyse a range of strategies for implementing secure design to prevent and mitigate cyberattacks.
Learning Outcome 4 4. Be able to apply cryptographic techniques to encrypt and decrypt information.	4.1 4.2 4.3 4.4 4.5 4.6	Describe key historic events leading up to the development of modern cryptographic techniques Outline basic illustrative cipher techniques. Compare encryption to hashing. Explain the value of cryptography for a range of cyber security applications. Explain the basic mathematical principles underlying encryption. Analyse a variety of encryption techniques.

	4.7	Use software to implement public and private key encryption of information and network connections.
Learning Outcome 5 5. Understand legal and societal issues concerning information security.	5.1 5.2 5.3	Explain the needs for legal and regulatory frameworks to govern Cyber Security and data protection. Analyse the effectiveness of legal frameworks in ensuring information security. Evaluate the trade-off between privacy and security in computer systems.

Unit 819 Principles of Computer Systems

UAN: D/651/4219
Level: 4
Credit value: 20
GLH: 100

Aim

This unit introduces the foundational concepts, methodologies, and techniques of computer systems and hardware. It focuses on the essential elements of how computers operate, covering topics such as logic design, state machines, assembly-level representation, parallelisation, and performance evaluation.
Students will explore system design both theoretically and practically, enhancing their understanding by implementing their own designs and critically analysing existing systems. By the end of the unit, students will have developed a deeper understanding of the different components of computer systems, and how they interact to perform information processing.

Learning outcome 1 1. Understand the architecture and key components of computer systems.	1.1	Describe the key hardware components of a computer system and their functions.
	1.2	Explain how data is processed and transferred within a computer.
	1.3	Explain how different components interact to improve system performance.
	1.4	Evaluate the benefits and impact of parallel processing on computing performance.
Learning outcome 2 2. Understand the types and functions of memory in computing systems.	2.1	Describe the Von Neumann computing paradigm.
	2.2	Compare the different forms of memory that computers use and their roles (primary, cache, and secondary).
	2.3	Analyse the impact of processor-memory separation on computing performance.
	2.4	Evaluate the difference between volatile and non-volatile memory in different computing contexts.
Learning outcome 3 3. Be able to design and implement simple state machines	3.1	Describe the principles of state machines and their role in computer systems.
	3.2	Design state diagrams for a range of processing scenarios.
	3.3	Implement basic state machines using tools.
	3.4	Critically evaluate the performance of state machines and propose improvements.

Learning outcome 4 4. Understand the fundamentals of logic circuits and gates.	4.1 Describe the representation of information in binary in computer systems. 4.2 Explain the common types of logic operation and corresponding logic gates. 4.3 Explain how logical operations are implemented using transistors in electronic circuits. 4.4 Explain how complex arithmetic operations can be built up from basic logic.
Learning outcome 5 5. Understand the fundamentals of assembly language and how it relates to higher level programming languages.	5.1 Describe the role of assembly language in performing low-level computer operations. 5.2 Explain the different assembly languages that exist and evaluate their differences. 5.3 Analyse how higher-level programming languages are abstracted from lower-level languages and the benefits of doing so. 5.4 Implement basic routines in ARM assembly language.

Unit 820 Algorithms and Data Structures

UAN: J/651/4220
Level: 4
Credit value: 20
GLH: 100

Aim

This module introduces data structures, algorithms, and the principles of computational complexity. Learners will become familiar with a variety of data structures and understand their similarities and differences. Both linear data structures such as arrays and lists, and non-linear data structures including graphs and trees, will be introduced. Time and space complexity, including notation for and analysis of complexity, are introduced as core considerations for algorithmic design. A range of popular sort, search, and pathfinding algorithms are introduced. Learners will gain practical experience in writing pseudocode for algorithms and using this to write, test, and debug programmed implementations of algorithms.

Learning outcome 1 1. Understand a range of essential data structures.	1.1 Describe a data structure. 1.2 Explain a range of linear and non-linear data structures. 1.3 Analyse the impact of the choice of data structure on algorithmic performance.
Learning Outcome 2 2. Understand algorithmic complexity and appreciate its importance.	2.1 Demonstrate an ability to write Big O and little o notation to describe algorithmic complexity. 2.2 Explain the impact of algorithmic design choice on computational complexity for a variety of examples. 2.3 Explain the difference between P and NP. 2.4 Evaluate open problems in problem classification and the impacts of future technologies on the difficulty in solving different classes algorithms.
Learning outcome 3 3. Understand a range of algorithmic techniques.	3.1 Use recursion and memoisation as tools to design solutions to problems. 3.2 Describe a range of sort, search, and pathfinding algorithms. 3.3 Explain the trade-off between memory and computational complexity in algorithmic design. 3.4 Critically analyse algorithmic paradigms such as the greedy

	approach and divide and conquer approach. 3.5 Evaluate sort, search, and pathfinding algorithms in terms of their space and time complexity.
Learning outcome 4 4. Be able to write algorithms in popular programming languages.	4.1 Explain how to write pseudo-code for algorithmic design. 4.2 Implement pseudocode in a programming language. 4.3 Evaluate the correctness and speed of an algorithm. 4.4 Use debugging to correct errors in algorithmic implementation.

Unit 821 Computer Networks

UAN: K/651/4221
Level: 4
Credit value: 20
GLH: 100

Aim

This unit introduces the principles of computer networking. Learners are introduced to network models and the different layers of computer networks. They will learn about the TCP/IP model and protocol suite, understanding the need for the separation between the different layers. A range of protocols are introduced, with a particular focus given to the key protocols of ethernet, IP and TCP/UDP. Learners will learn about the networking hardware that is essential for connecting devices in the physical world, including routers, switches. The unit concludes with the learners putting theory into practice by designing a practical computer network for a given setting.

Learning outcome 1 1. Understand the core principles of computer networks.	1.1 Describe the OSI and TCP/IP networking models and their layers and compare them. 1.2 Explain how different TCP/IP network protocols enable communication across devices and networks. 1.3 Analyse protocols and protocol suites.
Learning outcome 2 2. Understand how devices connect to form networks.	2.1 Describe the IP protocol. 2.2 Explain the role of network hardware components such as routers and switches. 2.3 Analyse the security measures used to secure communication across computer networks. 2.4 Evaluate mobile and wireless connection protocols and technologies.
Learning outcome 3 3. Be able to design a computer network suited to a given setting.	3.1 Decide on a network topology that is best suited to a given application and setting. 3.2 Analyse different network topologies and their role in disparate communication settings. 3.3 Design appropriate addressing schemes for a particular network. 3.4 Evaluate the best choice of network hardware for a given network.

Unit 822 Mathematics for Computer Science

UAN: L/651/4222
Level: 4
Credit value: 20
GLH: 100

Aim

In this unit, learners will become familiar with key subjects in mathematics that are of use in the field of computer science. The unit begins with an introduction to binary numbers, which form the basis of the representation of information in computers. Learners will then be taught key principles in the fields of Boolean propositional logic and set theory, including Boolean operators, and will be given an introduction to proof techniques in mathematics. Number theory, linear algebra, and sequences and series are also covered, to equip learners with a mathematical toolset that will enable them to reason about problems in computing.

Learning outcome 1 1. Understand binary and the representation of numbers in different bases.	1.1 Describe what is meant by binary representation. 1.2 Explain how to change between bases. 1.3 Explain how numbers are represented in binary. 1.4 Evaluate the use of binary representation in the field of computer science
Learning outcome 2 2. Understand the principles of propositional logic, set theory, and proof as applicable to computer science.	2.1 Describe a range of mathematical proof techniques. 2.2 Describe the possible algebraic operations in set theory. 2.3 Explain the key principles of (Boolean) propositional logic. 2.4 Use truth tables to detail the possible outcomes of binary logical operators. 2.5 Analyse the similarities between Boolean logical operators and Boolean algebra in set theory. 2.6 Evaluate mathematical proof techniques and the circumstances under which they may be appropriate.
Learning outcome 3 3. Understand key topics in number theory and their application in computer science.	3.1 Describe the key principles of modular arithmetic and finite fields. 3.2 Explain a range of key number theoretic theorems and proofs. 3.3 Analyse the differences between ordinary and modular arithmetic.

	3.4 Evaluate the use of modular arithmetic and prime numbers in security applications such as cryptography and hashing.
Learning outcome 4 4. Understand series and sequences and their importance in computer science.	4.1 Describe what is meant by a sequence and a series. 4.2 Explain the difference between arithmetic and geometric series and sequences. 4.3 Use appropriate notation to represent sequences
Learning outcome 5 5. Understand series and sequences and their importance in computer science.	5.1 Describe what is meant by vectors and matrices. 5.2 Use row-reduction to convert a matrix to echelon form. 5.3 Solve linear systems of equations using matrices. 5.4 Analyse the significance of the determinant of a matrix. 5.5 Use a range of techniques to decompose matrices and evaluate their usefulness.

Unit 823 Operating Systems

UAN: M/651/4223
Level: 4
Credit value: 20
GLH: 100

Aim

In this unit, learners will explore the role and principles of operating systems. Operating systems are introduced as software to manage resources in a computer and the need for operating systems is explained. The principal functions of operating systems are introduced, including memory management and process management, and learners will gain an appreciation for the trade-offs that exist in allocating hardware resources to processes. Learners will conclude by leveraging the understanding acquired through the unit to write their own operating system code at the kernel level.

Learning outcome 1 1. Understand the essential function and architecture of modern operating systems.	1.1 Outline a range of common operating systems for computers and mobile devices. 1.2 Describe the historical development and need for operating systems. 1.3 Explain the key components of an operating system. 1.4 Explain how operating systems manage hardware and software resources. 1.5 Analyse the different kinds of operating system that exist. 1.6 Evaluate common trade-offs in resource management for operating systems.
Learning outcome 2 2. Understand memory and I/O management techniques used by modern operating systems.	2.1 Compare different file system types. 2.2 Describe different approaches to memory management. 2.3 Describe the role of device drivers in facilitating communication with hardware devices. 2.4 Explain the role of virtual (swap) memory. 2.5 Explain the role of paging in allowing for non-contiguous memory allocation. 2.6 Analyse the difference between fixed and dynamic loading and linking strategies.

Learning outcome 3 3. Understand process management techniques used by modern operating systems.	3.1 Describe what is meant by a process. 3.2 Explain the contents of the process control block and its importance. 3.3 Explain the way that processes are represented for management by operating systems. 3.3 Analyse the range of responsibilities an operating system has for managing the execution of processes.
Learning outcome 4 4. Understand the principles of and potential problems with concurrency in operating systems.	4.1 Outline the difference between independent and cooperating processes. 4.2 Describe what is meant by concurrency. 4.3 Explain the memory sharing and message passing approaches to inter process communication. 4.4 Analyse deadlock and race conditions as potential problems in concurrent processing settings and propose strategies to mitigate them.
Learning outcome 5 5. Be able to write programs at the kernel level.	5.1 Develop kernel modules that interact with hardware devices and system resources. 5.2 Implement system calls to extend kernel functionality and interact with user-level applications. 5.3 Implement kernel-level programs to perform different tasks. 5.4 Use debugging to test kernel-level code to ensure stability and performance.